DIGITAL DOWNLOAD हिंदी

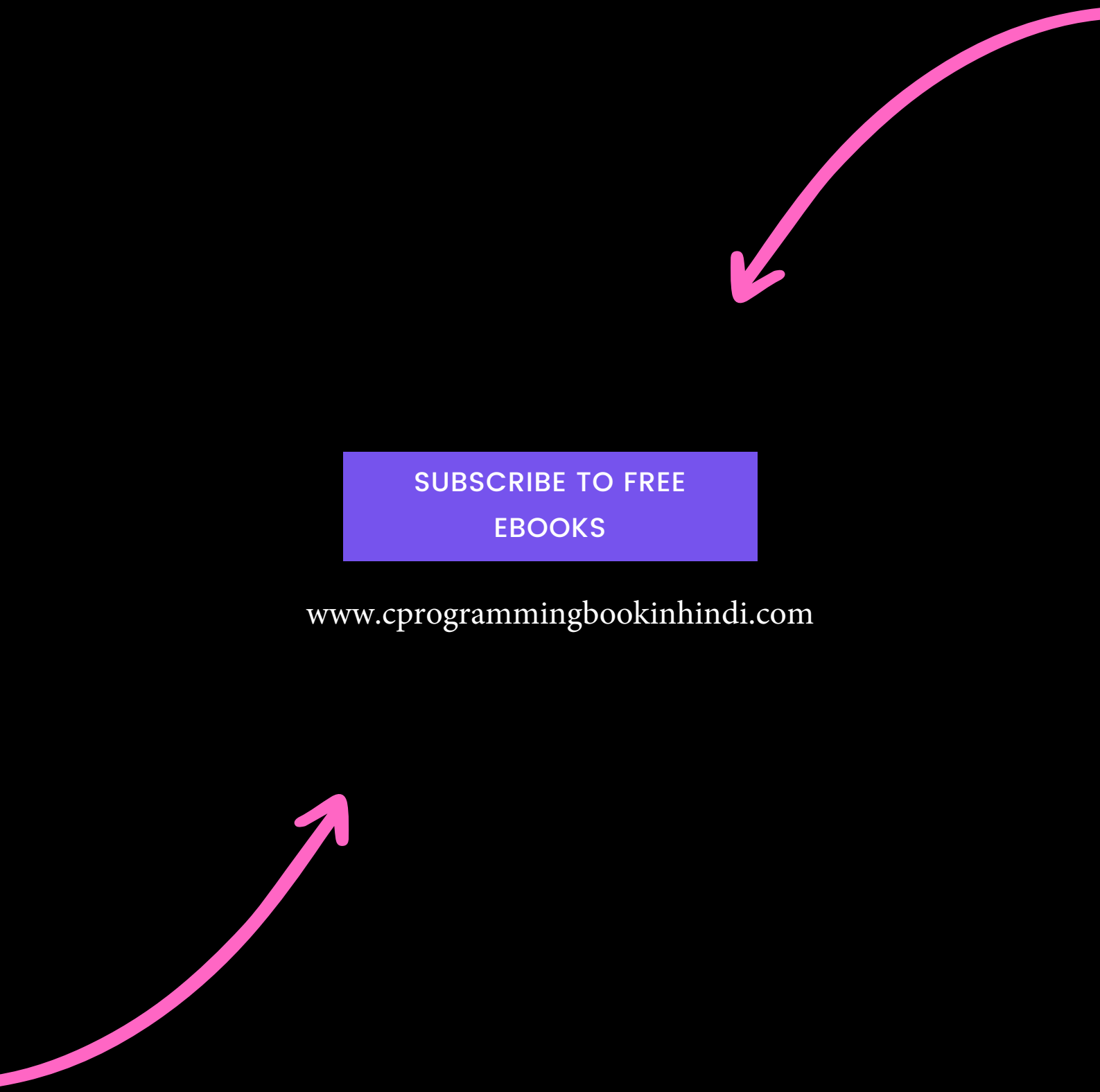
JavaScript for beginners to advance

Thanks for purchasing this Digital Download.

What's the success?

“We learn from failure, not from success!”
— Bram Stoker, Dracula

Please Keep this PDF in a safe place!


SUBSCRIBE TO FREE
EBOOKS
www.cprogrammingbookinhindi.com

JavaScript for Beginners to advance in Hindi

Copyright © 2017 by Machhindra Kalan

All rights reserved. No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without the prior written permission of the copyright owner and the publisher.

Trademarked names may appear in this book. Rather than use a trademark symbol with every occurrence of a trademarked name, we use the names only in an editorial fashion and to the benefit of the trademark owner, with no intention of infringement of the trademark.

Lead Editor: Machhindra Kalan

Distributed to the book trade worldwide by Web Study Lab Pvt Ltd, Location - Mumbai, Palava city Pin 421204

E-mail: machhi29@gmail.com, or

Visit: www.cprogrammingbookinhindi.com

For information on translations, please contact Web Study Lab Pvt Ltd, Location - Mumbai, Palava city Pin 421204

Phone +91 86555-67778

The information in this book is distributed on an “as is” basis, without warranty. Although every precaution has been taken in the preparation of this work, the author shall not have any liability to any person or entity with respect to any loss or damage caused or alleged to be caused directly or indirectly by the information contained in this book.

www.cprogrammingbookinhindi.com
3

This book
is dedicated
to those

who really wants
to be a

JavaScript for Beginners to advance

INDEX OF CONTENTS

Table of Contents

Introduction.....9

Chapter 1: The World's Most Misunderstood Programming Language . . .13

What Is JavaScript?.....14
The Eich-man cometh.....14
Mocha-licious.....15
We need more effects!.....16
JavaScript grows up.....17
Dynamic scripting language.....17
What Does JavaScript Do?.....18
Why JavaScript?.....20
JavaScript is easy to learn.....23
Where is JavaScript? JavaScript is everywhere!.....24
JavaScript is powerful!.....27
JavaScript is in demand.....28

Chapter 2: Writing Your First JavaScript Program 29

Setting Up Your Development Environment.....30
Downloading and installing Chrome.....32
Downloading and installing a code editor.....33
Reading JavaScript Code.....35
Running JavaScript in the Browser Window.....36
Using JavaScript in an HTML event attribute.....37
Using JavaScript in a script element.....39
Including external JavaScript files.....40
Using the JavaScript Developer Console.....46
Commenting your code.....47

Chapter 3:Working with Variables 55

Understanding Variables 56
Declaring Variables 56
Understanding Global and Local Scope....57
Naming Variables 57
Creating Constants Using the const Keyword....58
Working with Data Types....59
Number data type 60
String data type 63
Boolean data type 65
NaN data type....69
undefined data type....70

Chapter 4: Understanding Arrays 73

Making a List....75
Array Fundamentals....76
Arrays are zero indexed 77
Arrays can store any type of data....78
Creating Arrays 79
Using the new keyword method 79
Array literal....79
Populating Arrays 80
Understanding Multidimensional Arrays 80
Accessing Array Elements 82
Looping through arrays....83
Array properties 83
Array methods 84
Using array methods....87

Chapter 5: Working with Operators, Expressions, and Statements 88

Express Yourself....88
Hello, Operator....90
Operator precedence 92
Types of Operators 92
Assignment operators 92
Comparison operators 93
Arithmetic operators 93
String operator 95
Bitwise operators 95
Logical operators....97
Special operators....88
Combining operators....100

Chapter 6: Getting into the Flow with Loops and Branches 104

Branching Out....105
if . . . else....106
Switch....107
Here We Go: Loop De Loop....107
for....108
for . . . in....110
while loops....113
do . . . while....116
break and continue....118

Chapter 7: Getting Functional 119

Understanding the Function of Functions....119
Using Function Terminology....119
Define a function....119
Function head....121
Function body....122
Call a function....123
Defining parameters and passing arguments....125
Return a value....126
The Benefits of Using Functions....127
Writing Functions....127
Returning Values....128
Passing and Using Arguments....128
Passing arguments by value....130
Passing arguments by reference....131
Calling a function without all of the arguments....132
Setting default parameter values....133
Calling a function with more argument than parameters....135
Getting into arguments with the arguments object....135
Function Scope....136
Anonymous Function....137
Knowing the differences between anonymous and named functions
Self-executing anonymous functions....138
Do it Again with Recursion....139
Functions within Functions....140

Chapter 8: Making and Using Objects 141

Object of My Desire 141
Creating Objects 142
Defining objects with object literals 143
Defining objects with an Object constructor 145
Retrieving and Setting Object Properties 146
Dot notation 146
Square bracket notation 147
Deleting Properties 147
Working with Methods 149
Using this 150
An Object-Oriented Way to Become Wealthy: Inheritance 152
Constructing Objects with constructor functions 155
Modifying an object type 156
Creating Objects with Object.create 157

Chapter 9: Controlling the Browser with the Window Object 158

Understanding the Browser Environment 158
The user interface 159
Loader 161
HTML parsing 163
CSS parsing 166
JavaScript parsing 166
Layout and rendering 167
Igniting the BOM 169
The Navigator object 172
The Window object 173
Using the `Window` object's methods 175

Chapter 10: Manipulating Documents with the DOM 175

Understanding the DOM 177
Node Relationships 179
Using the Document Object's Properties and Methods 180
Using the Element Object's Properties and Methods 181
Working with the Contents of Elements 182
innerHTML 185
Setting attributes 186
Getting Elements by ID, Tag Name, or Class 186
getElementById 188
getElementsByTagName 190
getElementsByClassName 193

Using the Attribute Object's Properties 195
Creating and appending elements 195
Removing elements 196

Chapter 11: Using Events in JavaScript 197

Knowing Your Events 198
Handling Events 198
Using inline event handlers 205
Event handling using element properties 207
Event handling using addEventListener 208
Stopping propagation 210

Chapter 12: Integrating Input and Output 211

Understanding HTML Forms 211
The form element 213
The label element 213
The input element 204
The select element 205
The textarea element 206
The button element 210
Working with the Form Object 212
Using Form properties 215
Using the Form object's methods 218
Accessing form elements 223
Getting and setting form element values 225
Validating user input 228

Chapter 13: Working with CSS and Graphics 230

Using the Style Object 231
Getting the current style of an element 231
Setting style properties 232
Animating Elements with the Style Object 235
Working with Images 237
Using the Image object 239
Creating rollover buttons 241
Grow images on mouseover 245
Creating an image slideshow 246
Using the Style Object's Animation Properties 248

Chapter 14: Searching with Regular Expressions 250

Finding It Out with Regular Expressions 252
Writing Regular Expressions 255
Using the RegExp object 256

Regular expression literals ... 260
Testing regular expressions ... 261
Special character in regular expressions ... 263
Using Modifiers ... 263
Coding with Regular Expressions ... 265

Chapter 15: Understanding Callbacks and Closures 265

What Are Callbacks? ... 265
Passing functions as arguments ... 266
Writing functions with callbacks ... 267
Using named callback functions ... 270
Understanding Closures ... 273
Using Closures ... 275

Chapter 16: Embracing AJAX and JSON 278

Working Behind the Scenes with AJAX ... 279
AJAX examples ... 280
Viewing AJAX in action ... 283
Using the `XMLHttpRequest` object ... 285
Working with the same-origin policy ... 288
Using CORS, the silver bullet for AJAX requests ... 292
Putting Objects in Motion with JSON ... 295

Chapter 17: HTML5 APIs 297

Understanding How APIs Work ... 299
Checking HTML5 API browser support ... 301
Getting to know HTML5's APIs ... 303
Using Geolocation ... 307
What does geolocation do? ... 309
How does geolocation work? ... 311
How do you use geolocation ... 313
Combining geolocation with Google maps ... 314
Accessing Audio and Video ... 315

Chapter 18: jQuery 316

Writing More and Doing Less ... 317
Getting Started with jQuery ... 319
The jQuery Object ... 322
Is Your Document Ready? ... 325
Using jQuery Selectors ... 328
Changing Things with jQuery ... 330

Getting and setting attributes....330
Changing CSS....330
Manipulating elements in the DOM....331
Events....331
Using on( ) to attach events....332
Detaching with off( )....332
Binding to events that don't exist yet....333
Other event methods....335
Effects....335
Basic effects....335
Fading effects....336
Sliding effects....337
Setting arguments for animation methods....337
Custom effects with animate( )....337
Playing with jQuery animations....337
AJAX....338
Using the ajax( ) method....338
Shorthand AJAX methods....**338**

Chapter 19: Ten JavaScript Frameworks and Libraries..... 338

Angular JS....339
Backbone.js....340
Ember.js....342
Famo.us....343
Knockout....345
QUnit....345
underscore.js....347
Modernizr....348
Handlebars.js....349
jQuery....350

Chapter 20: Ten Common JavaScript Bugs and How to Avoid Them. .351

Equality Confusion....351
Avoiding misuse of assignment....352
Dodging the equals pitfalls....353
Mismatched Brackets....353
Mismatched Quotes....355
Missing Parentheses....355
Missing Semicolon....356
Capitalization Errors....357

Referencing Code Before It's Loaded 357
Bad Variable Names 358
Scope Errors 360
Missing Parameters in Function Calls 361
Counting Errors: Forgetting That JavaScript Counts from 0 361

Chapter 21: Ten Online Tools to Help You Write Better JavaScript 362

JSLint 363
JSFiddle.net 364
JSBin 365
javascriptcompressor.com 366
jsbeautifier.org 367
JavaScript RegEx generator 368
JSONformatter 369
jshint.com 370
Mozilla Developer Network 371
Douglas Crockford 372

INTRODUCTION

जावासक्रिप्ट एक सक्रिप्टिंग भाषा है जो आपको गतिशील, व्यक्तिगत और इंटरैक्टिव सामग्री प्रदान करके स्थिर वेब applications को बढ़ाने में सक्षम बनाती है। यह आपकी साइट पर आगंतुकों के अनुभव को बेहतर बनाता है और यह अधिक expressions है कि वे फिर से यात्रा करेंगे। आपने आकर्षक ड्रॉप–डाउन मेनू, चलते हुए पाठ और बदलते सामग्री को देखा होगा जो अब वेब साइटों पर व्यापक हैं – वे जावासक्रिप्ट के साथ सक्षम हैं। सभी प्रमुख ब्राउज़रों द्वारा समर्थित, जावासक्रिप्ट वेब पर पसंद की भाषा है। यह भी हो सकता है। उदाहरण के लिए, प्रशासनिक कार्यों को स्वचालित करने के लिए वेब एप्लिकेशन के बाहर उपयोग किया जाता है।

इस पुस्तक का उद्देश्य आपको उन सभी को पढ़ाना है जो आपको जावासक्रिप्ट के साथ प्रयोग करना शुरू करने के लिए जानने की आवश्यकता है: यह क्या है, यह कैसे काम करता है, और आप इसके साथ क्या कर सकते हैं। मूल सिंटैक्स से शुरू करके, आप सीखेंगे कि शक्तिशाली वेब एप्लिकेशन कैसे बनाएं। चिंता न करें अगर आपने पहले कभी प्रोग्राम नहीं किया है – तो यह किताब सिखाएगी।

आप सभी को जानने की जरूरत है, कदम से कदम। आप पाएंगे कि जावासक्रिप्ट प्रोग्रामिंग की दुनिया के लिए एक महान परिचय हो सकता है: ज्ञान और समझ के साथ जो आप इस पुस्तक से प्राप्त करेंगे, आप कंप्यूटिंग की दुनिया में नई और अधिक उन्नत तकनीकों को सीखने के लिए आगे बढ़ने में सक्षम हो।

आपको इस पुस्तक का उपयोग करने की आवश्यकता क्या है? क्योंकि जावासक्रिप्ट एक पाठ–आधारित तकनीक (**text-based technology**) है, आप सभी को वास्तव में जावासक्रिप्ट नोटपैड या नोटपैड़ ++, वाले दस्तावेज़ बनाने की आवश्यकता है।

साथ ही, इस पुस्तक में कोड को आज़माने के लिए, आपको एक वेब ब्राउज़र की आवश्यकता होगी जो जावास्क्रिप्ट के आधुनिक संस्करण का समर्थन करता है। आदर्श रूप से, इसका मतलब इंटरनेट एक्सप्लोरर **6** या उसके बाद और फ़ायरफ़ॉक्स **1.5** है या बाद में। किताब में है इन दो ब्राउज़रों के साथ बड़े पैमाने पर परीक्षण किया गया है। हालाँकि, कोड को अधिकांश आधुनिक वेब ब्राउज़र में काम करना चाहिए, हालाँकि बाद के अध्यायों में कुछ कोड, जहाँ आप डायनामिक HTML और स्क्रिप्टिंग की जाँच करते हैं। DOM, विशेष ब्राउज़र के लिए विशिष्ट है; लेकिन प्रस्तुत कोड का अधिकांश हिस्सा क्रॉस–ब्राउज़र है। जहां अपवाद हैं, उन्हें स्पष्ट रूप से नोट किया जाएगा।

PART I

GETTING STARTED WITH JAVASCRIPT

CHAPTER 1

THE WORLD'S MOST MISUNDERSTOOD PROGRAMMING LANGUAGE

जावासक्रिप्ट को हमेशा उतना उच्च नहीं माना जाता जितना आज है। कुछ लोगों ने इसे दुनिया की सबसे अच्छी और सबसे खराब प्रोग्रामिंग भाषा कहा है। पिछले कुछ वर्षों में, प्रोग्रामर ने दुभाषिए जावासक्रिप्ट और जावासक्रिप्ट को लिखने के तरीके में काफी सुधार किया है। इन सुधारों ने आज जावासक्रिप्ट को पहले की तुलना में बहुत बेहतर भाषा बना दिया है।

इस अध्याय में, आपको पता चलता है कि जावासक्रिप्ट क्या है और भाषा का थोड़ा सा इतिहास। आपको यह भी पता चलता है कि जावासक्रिप्ट क्या करता है और आपको इसे जानने की आवश्यकता क्यों है।

इस अध्याय से संबंधित ऑनलाइन अभ्यासों की जाँच करने के लिए वेबसाइट पर जाना न भूलें।

What Is JavaScript?

वेब के शुरुआती दिनों में, वेब पेजों के लिए ब्राउज़र सरल पाठक थे (चित्र देखें 1-1) उनके पास वस्तुतः कोई क्षमता नहीं थी, केवल विभिन्न आकार के फोंट में पाठ प्रदर्शित करने की क्षमता को छोड़कर। जैसे ही Microsoft ने अपना इंटरनेट एक्सप्लोरर ब्राउज़र जारी किया, ब्राउज़र वॉर चालू हो गए और सुविधाएँ उड़ने लगीं! एक ब्राउज़र ने छवियों को प्रदर्शित करने की क्षमता का परिचय दिया, फिर दूसरे ने अलग–अलग फोंट लगाने की क्षमता का परिचय दिया, और फिर ब्लिंकिंग टेक्स्ट, मूविंग टेक्स्ट और अन्य सभी प्रकार की निराला क्षमताओं को पेश किया गया!

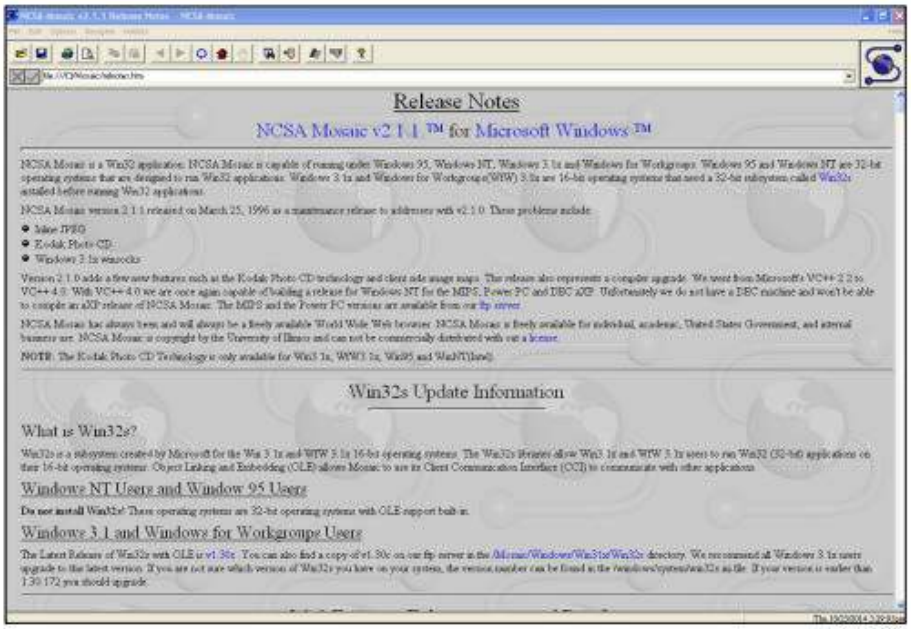


Figure 1-1: The first web browsers weren't much to look at.

यह बहुत पहले नहीं था कि किसी को यह विचार मिले कि ब्राउज़र वास्तव में उपयोगी चीजें कर सकते हैं, बजाय केवल फैंसी दस्तावेज़ प्रदर्शन कार्यक्रमों के अभिनय के।

The Eich-man cometh

1995 में जावास्क्रिप्ट ने नेटस्केप में अपनी शुरुआत की। जावास्क्रिप्ट के निर्माता, ब्रैंडन ईच, ने रिकॉर्ड समय में जावास्क्रिप्ट लिखा था (कुछ कहते हैं कि दस दिनों में!) विभिन्न अन्य प्रोग्रामिंग भाषाओं से कई बेहतरीन सुविधाओं को उधार लेकर। बाजार की भीड़ ने भाषा के डिजाइन में कुछ दिलचस्प विचित्रता (या, कम विनम्रता से वर्णित, गलतियाँ) पैदा की। इसका परिणाम एक तरह का स्क्रीनशॉट है जो भाषा की तरह है जो लोगों को भ्रामक रूप से परिचित करता है जो अन्य प्रोग्रामिंग भाषाओं के साथ अनुभवी हैं।

Mocha-licious

जावास्क्रिप्ट का मूल नाम मोचा था। नेटस्केप नेविगेटर के पहले बीटा परिनियोजन के साथ इसका नाम बदलकर लाइवस्क्रिप्ट रखा गया और

फिर इसे बदल दिया गया। जावासक्रिप्ट जब इसे नेटस्केप 2 ब्राउज़र में 1995 बनाया गया था। Microsoft ने बहुत तेज़ी से reverse इंजिन को उल्टा किया और इंटरनेट एक्सप्लोरर में इसका सटीक क्लोन पेश किया, इसे ट्रेडमार्क मुद्दों के बारे में जानने के लिए इसे Jscript कहा।

नेटस्केप ने एक्मा इंटरनेशनल के नाम से जाने जाने वाले मानक संगठन में जावासक्रिप्ट जमा किया और इसे EMCAScript के रूप में अपनाया और मानकीकृत 1997 किया गया।

जावासक्रिप्ट के निर्माता ब्रैंडन ईच ने मानकीकृत भाषा के नाम के बारे में टिप्पणी की; यह बताते हुए कि ECMAScript एक अवांछित व्यापार नाम था जो एक त्वचा रोग की तरह लगता है।

न केवल ECMAScript एक प्रोग्रामिंग भाषा के लिए एक अनपेक्षित नाम है, नेटस्केप द्वारा भाषा को दिया गया नाम और जिसे ज्यादातर लोग संदर्भित करते हैं ECMAScript के रूप में यह है, बल्कि दुर्भाग्यपूर्ण भी है। यदि आप पहले से ही जानते हैं कि जावा में कैसे प्रोग्राम करना है या यदि आप किसी बिंदु पर सीखते हैं, तो इसे रखना बहुत अच्छा विचार है। ध्यान रखें कि दो भाषाओं में कुछ समानताएं हो सकती हैं, लेकिन वे वास्तव में, काफी अलग भाषा हैं।

We need more effects!

जब जावासक्रिप्ट की शुरुआत हुई, तो यह वेब पृष्ठों को अधिक गतिशील बनाने के तरीके के रूप में बहुत लोकप्रिय हो गया। इसलिए - डायनेमिक HTML (DHTML) कहा जाता है। जावासक्रिप्ट का परिणाम वेब ब्राउज़र में बनाया जा रहा है, और इसने सभी प्रकार के मजेदार प्र expressionsों को सक्षम किया है, जैसे गिरते हुए हिमपात प्र expressions (चित्र 1-2 देखें), पॉप अप विंडोज़, और वेब पेज कोनों को कर्लिंग करना, लेकिन ड्रॉप–डाउन मेनू और फ़ॉर्म सत्यापन जैसी और भी उपयोगी चीजें।

JavaScript grows up

अब अपने तीसरे दशक में प्रवेश करते हुए, जावास्क्रिप्ट दुनिया की सबसे अधिक उपयोग की जाने वाली प्रोग्रामिंग भाषा बन गई है और लगभग हर पर्सनल कंप्यूटर में दुनिया में इस पर कम से कम एक ब्राउज़र है जो जावास्क्रिप्ट कोड चला सकता है।

जावास्क्रिप्ट इतना लचीला है कि इसका इस्तेमाल किया जा सकता है और इसे नॉनप्रोग्रामर्स द्वारा सीखा जा सकता है, लेकिन इतना शक्तिशाली कि यह पेशेवर द्वारा इस्तेमाल किया जा सकता है (और है) प्रोग्रामर आज इंटरनेट पर लगभग हर वेबसाइट पर कार्यक्षमता को सक्षम करने के लिए, single (single) पृष्ठ (page) साइटों से लेकर Google, अमेज़ॅन, फेसबुक जैसी विशाल साइटों तक, और कई, कई अन्य!

Dynamic scripting language

जावास्क्रिप्ट को अक्सर एक गतिशील स्क्रिप्टिंग भाषा के रूप में वर्णित किया जाता है। इसका क्या अर्थ है यह समझने के लिए, हमें पहले कुछ शब्दों को परिभाषित करने और कुछ संदर्भ प्रदान करने की आवश्यकता है।

जावास्क्रिप्ट के बारे में आम गलतफहमी

इन वर्षों में, जावास्क्रिप्ट में इसके बारे में कही गई कुछ बेहद घटिया बातें हैं। जबकि कभी–कभी अफवाहें दिलचस्प होती हैं, वे हमेशा सच नहीं होते हैं।

निम्न सूची जावास्क्रिप्ट के बारे में कुछ सामान्य गलतफहमी बताती है:

- Myth: जावास्क्रिप्ट एक वास्तविक प्रोग्रामिंग भाषा नहीं है। वास्तविकता: जावास्क्रिप्ट का उपयोग अक्सर वेब ब्राउज़र में तुच्छ कार्यों के लिए किया जाता है, लेकिन यह इसे प्रोग्रामिंग भाषा से कम नहीं बनाता है। वास्तव में, जावास्क्रिप्ट में कई उन्नत

विशेषताएं हैं जिन्होंने प्रोग्रामिंग भाषाओं के लिए बार उठाया है और अब PHP, C, और यहां तक कि जावा जैसी भाषाओं में भी नकल की जा रही है।

- **Myth**: जावास्क्रिप्ट जावा से संबंधित है। हकीकत: नहीं। जावास्क्रिप्ट का नाम विशुद्ध रूप से एक विपणन रणनीति के रूप में आविष्कार किया गया था क्योंकि जावा उस समय अविश्वसनीय रूप से लोकप्रिय था जब जावास्क्रिप्ट बाहर आया था।

- **Myth:** जावास्क्रिप्ट नया है। वास्तविकता: जावास्क्रिप्ट 20 से अधिक वर्षों के लिए किया गया है! कुछ पेशेवर जावास्क्रिप्ट प्रोग्रामर जिन्हें हम जानते हैं कि जावास्क्रिप्ट के निर्माण के समय भी पैदा नहीं हुए थे।

- Myth: जावास्क्रिप्ट छोटी गाड़ी है और विभिन्न ब्राउज़रों में अलग तरह से चलती है। वास्तविकता: जबकि यह कुछ मामलों में सही हुआ करता था, ब्राउज़र निर्माताओं ने बहुत पहले जावास्क्रिप्ट के मानकीकृत संस्करण का समर्थन करने का फैसला किया। हर ब्राउज़र आज ही JavaScript चलाएगा कंप्यूटर प्रोग्राम निर्देशों का समूह हैं जो कंप्यूटर को काम करने का कारण बनाते हैं। हर कंप्यूटर प्रोग्रामिंग भाषा में निर्देशों का एक सेट होता है और एक निश्चित तरीका होता है कि मनुष्य को उन निर्देशों को लिखना होगा। कंप्यूटर इन निर्देशों को सीधे समझ नहीं सकता है।

एक प्रोग्रामिंग भाषा को समझने के लिए कंप्यूटर के लिए, उसे एक रूपांतरण प्रक्रिया से गुजरना पड़ता है जो मानव के पठनीय (और लिखने योग्य) निर्देशों को मशीनी भाषा में अनुवाद करता है। यह अनुवाद कब होता है, इसके आधार पर, प्रोग्रामिंग भाषाओं को मोटे तौर पर दो प्रकारों में विभाजित किया जा सकता है: Compiled और Interpreted की गई (चित्र 1-3 देखें)।

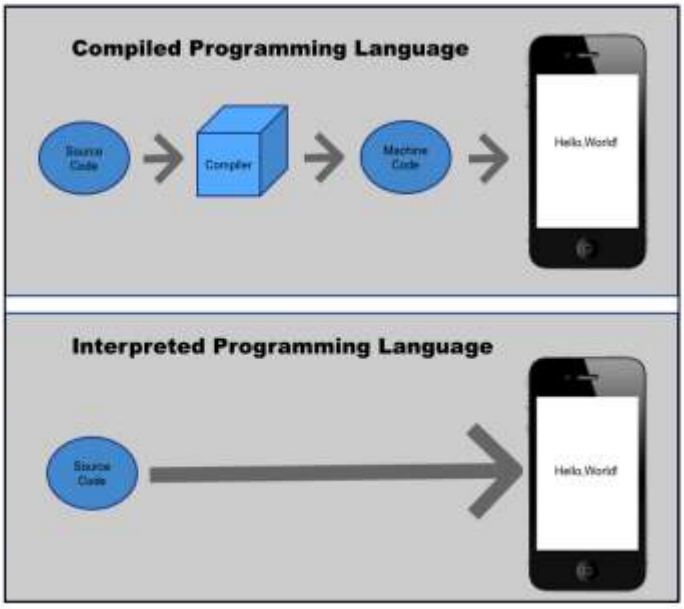


Figure 1-3: Programming languages are classified according to when the compilation takes place.

Compiled programming languages

Compiled प्रोग्रामिंग भाषाएं ऐसी भाषाएं हैं जिनमें एक प्रोग्रामर को कोड लिखना होगा और फिर उसे एक विशेष प्रोग्राम के माध्यम से चलाना होगा जिसे कंपाइलर कहा जाता है जो दिए गए कोड की Interpreted करता है और फिर इसे मशीन भाषा में परिवर्तित करता है। कंप्यूटर फिर Compiled प्रोग्राम को निष्पादित कर सकता है।

Compiled भाषाओं के उदाहरणों में C, C++ शामिल हैं। फोरट्रान, जावा, ऑब्जेक्टिव-C, और COBOL।

Interpreted programming languages

Interpreted की गई भाषाएं तकनीकी रूप से अभी भी कंप्यूटर द्वारा मशीनी भाषा में Compiled की जाती हैं, लेकिन संकलन उपयोगकर्ता के वेब ब्राउजर द्वारा सही तरीके से होता है क्योंकि प्रोग्राम चलाया जा रहा है। प्रोग्रामर जो Interpreted भाषा लिखते हैं, उन्हें चलाने के लिए कंप्यूटर को सौंपने से पहले अपने कोड को Compiled करने के चरण से गुजरने की आवश्यकता नहीं है।

एक Interpreted की गई भाषा में प्रोग्रामिंग का लाभ यह है कि किसी भी समय कार्यक्रम में बदलाव करना आसान है। हालांकि, नकारात्मक पक्ष

यह है कि संकलन कोड के रूप में इसे चलाया जा रहा है प्रक्रिया में एक और कदम बनाता है और कार्यक्रमों के प्रदर्शन को धीमा कर सकता है। आंशिक रूप से इस प्रदर्शन कारक के कारण, Interpreted की गई भाषाओं ने गंभीर प्रोग्रामिंग भाषाओं से कम होने के लिए प्रतिष्ठा प्राप्त की है।

हालाँकि, टाइम कंपाइलर्स और तेज़ कंप्यूटर प्रोसेसर में बेहतर just के कारण, यह धारणा तेजी से बदल रही है। इस संबंध में जावासक्रिप्ट का बड़ा expressions हो रहा है।

Interpreted की गई प्रोग्रामिंग भाषाओं के उदाहरणों में PHP, पर्ल, हास्केल, रूबी और निश्चित रूप से जावासक्रिप्ट शामिल हैं।

What Does JavaScript Do?

यदि आप वेब का उपयोग करते हैं, तो आप हर समय जावासक्रिप्ट का उपयोग कर रहे हैं। जावासक्रिप्ट के साथ सक्षम किया जा सकता है कि चीजों की सूची व्यापक है और से लेकर है।

जब आप Google एप्लिकेशन या फ़ेसबुक जैसे जटिल applications में फ़ॉर्म पर आवश्यक फ़ील्ड भरना भूल जाते हैं, तो आपको प्राप्त होने वाली साधारण सूचनाएँ। यहाँ वेब पर जावासक्रिप्ट के लिए सबसे आम उपयोग की एक छोटी सूची है:

- ✓ Nifty effects
- ✓ Input validation
- ✓ Rollover effects
- ✓ Drop-down/fly-out menus
- ✓ Drag and drop features
- ✓ Infinitely scrolling web pages
- ✓ Autocomplete
- ✓ Progress bars
- ✓ Tabs within web pages
- ✓ Sortable lists
- ✓ Magic Zoom (see Figure 1-4)

Figure 1-4: So-called Magic Zoom effects are enabled using JavaScript.

Why JavaScript?

जावास्क्रिप्ट वेब के लिए गतिशील उपयोगकर्ता इंटरफेस बनाने के लिए मानक बन गया है। जब भी आप एनिमेशन के साथ किसी वेब पेज पर जाते हैं, तब तक बहुत सुंदर है डेटा, एक बटन जो आपके ऊपर मंडराने पर बदल जाता है, या एक ड्रॉप डाउन मेनू, जावास्क्रिप्ट काम पर है। इसकी शक्ति और किसी भी वेब में चलने की क्षमता के कारण ब्राउज़र, जावास्क्रिप्ट कोडिंग एक आधुनिक वेब डेवलपर के लिए सबसे लोकप्रिय और आवश्यक कौशल है।

JavaScript is easy to learn

यह ध्यान रखें कि प्रोग्रामिंग भाषाएं लोगों को कंप्यूटर से बात करने का एक सरल तरीका देने और उन्हें यह बताने के लिए बनाई गई थीं कि उन्हें क्या करना है।

मशीन की भाषा की तुलना में, कंप्यूटर की सीपीयू जो भाषा बोलती है,

हर प्रोग्रामिंग भाषा आसान और समझ में आती है। आपको इस बात का एक नमूना देने के लिए कि आपका कंप्यूटर वास्तव में किस तरह के निर्देशों का पालन कर रहा है, **"Hello World"** लिखने के लिए यहां एक मशीन भाषा प्रोग्राम है।

```
b8    21 0a 00 00
a3    0c 10 00 06
b8    6f 72 6c 64
a3    08 10 00 06
b8    6f 2c 20 57
a3    04 10 00 06
b8    48 65 6c 6c
a3    00 10 00 06
b9    00 10 00 06
ba    10 00 00 00
bb    01 00 00 00
b8    04 00 00 00
cd    80
b8    01 00 00 00
cd    80
```

अब एक तरह से आप जावासक्रिप्ट के साथ इस सरल कार्य को पूरा कर सकते हैं:

```
alert("Hello World");
```

बहुत आसान है, हाँ?

एक बार जब आप सड़क के मूल नियमों (सिंटैक्स कहा जाता है) को सीखते हैं, जैसे कि parentheses brackets का उपयोग कब करना है और कब curly brackets ({}) का उपयोग करना है, तो जावासक्रिप्ट वास्तव में सादे पुरानी अंग्रेजी जैसा दिखता है।

प्रोग्रामिंग भाषा सहित किसी भी भाषा को सीखने में पहला कदम है, आपके शुरू होने के डर पर काबू पाना। जावासक्रिप्ट यह आसान बनाता है। वेब पर जावासक्रिप्ट कोड के हजारों सैंपल बिट्स हैं जिन्हें कोई भी उठा सकता है और उसके साथ खिलवाड़ करना शुरू कर सकता है। आपके पास पहले से ही आपके पास आवश्यक सभी उपकरण हैं (अध्याय 2 देखें), और जावासक्रिप्ट के साथ छोटे से शुरू करना आसान है और धीरे–धीरे महान और अद्भुत चीजें बनाने के लिए।

Where is JavaScript? JavaScript is everywhere!

सब जगह है! हालाँकि जावास्क्रिप्ट को मूल रूप से वेब ब्राउज़र में उपयोग करने के लिए डिज़ाइन किया गया था, फिर भी इसने कई अन्य स्थानों पर एक घर पाया है। आज, जावास्क्रिप्ट स्मार्टफोन और टैबलेट पर, वेब सर्वर पर, डेस्कटॉप applications में और सभी प्रकार के पोर्टेबल उपकरणों पर चलता है।

JavaScript in the web browser

जावास्क्रिप्ट खोजने के लिए सबसे आम जगह है, और यह मूल रूप से क्या करने के लिए डिज़ाइन किया गया था, वेब ब्राउज़र में चल रहा है। जब जावास्क्रिप्ट इस तरह से चलता है, तो इसे क्लाइंट। साइड जावास्क्रिप्ट कहा जाता है।

क्लाइंट - साइड जावास्क्रिप्ट वेब पेजों में अन्तरक्रियाशीलता जोड़ता है। यह कई मायनों में इसे पूरा करता है:

- ब्राउज़र को स्वयं नियंत्रित करने या ब्राउज़र की कार्यक्षमता का उपयोग करने से।
- वेब पेजों की संरचना और सामग्री में हेरफेर करके।
- वेब पृष्ठों की शैलियों (जैसे फोंट और लेआउट) में हेरफेर करके।
- अन्य स्रोतों से डेटा का उपयोग करके।

यह समझने के लिए कि जावास्क्रिप्ट वेब पृष्ठों की संरचना और शैली में हेरफेर करने में कैसे सक्षम है, आपको HTML5 और CSS3 के बारे में थोड़ा जानने की आवश्यकता है।

HTML5

हाइपरटेक्स्ट मार्कअप लैंग्वेज (एचटीएमएल) वह भाषा है जिसका उपयोग वेब पेजों की संरचना के लिए किया जाता है।

यह वेब ब्राउज़रों को सामग्री के बारे में जानकारी देने के लिए सामग्री (पाठ और चित्र) को चिह्नित करके काम करता है, जैसे कि क्या शीर्षक है, एक पैराग्राफ क्या है, जहां एक छवि जाती है, और इसी तरह। 1–1 लिस्टिंग एक सरल HTML दस्तावेज़ दिखाती है। चित्र 1–5 दिखाता है कि वेब ब्राउज़र इस दस्तावेज़ को कैसे प्रदर्शित करता है।

Listing 1-1: A Simple HTML Document

```
<!DOCTYPE html>
<html>
<head>
 <title>Hello, HTML!</title>
</head>
<body>
 <h1>This is HTML</h1>
 <p id="introduction">This simple document was written
          with Hypertext Markup Language.</p>
</body>
</html>
```

जावास्क्रिप्ट सीखने के साथ आगे बढ़ने के लिए अभी आपको HTML के बारे में जानने की जरूरत है:

- HTML में, brackets से घिरे वर्णों को टैग कहा जाता है।
- Marked समाप्त होने वाला टैग (जो सामग्री को चिह्नित किए जाने के बाद आता है) में पहले brackets के बाद एक स्लैश है। उदाहरण के लिए </p> एक समाप्त होने वाला टैग है।
- (दो टैग (आरंभ और अंत) का एक समूह, प्लस उनके बीच की सामग्री, एक element कहलाता है।
- Organized element आम तौर पर एक पदानुक्रमित तरीके से व्यवस्थित होते हैं (elements के भीतर निहित element)।
- elements में नाम / मान जोड़े हो सकते हैं, जिन्हें विशेषता कहा जाता है। यदि किसी element में विशेषताएँ हैं, तो वे शुरुआत

टैग में जाते हैं। नाम / मूल्य जोड़े उन दोनों के बीच एक समान चिन्ह लगाकर, quotes में, नाम के लिए (जो quotes में नहीं हैं) मान प्रदान करते हैं। उदाहरण के लिए, निम्नलिखित टैग में, चौड़ाई और ऊंचाई दोनों element के गुण हैं:

```
<div width="100" height="100"></div>
```

Ending कुछ elements में सामग्री नहीं होती है और इसलिए उन्हें समाप्त होने वाले टैग की आवश्यकता नहीं होती है। उदाहरण के लिए, img टैग, जो बस एक छवि को वेब पेज में सम्मिलित करता है, इस तरह दिखता है:

```
<img src="myimage.jpg" width="320" height="200"
alt="Here is a picture of my dog.">
```

Figure 1-5: Web browsers use HTML to render web pages.

छवि दिखाने के लिए आवश्यक सभी डेटा विशेषताओं का उपयोग करके शुरुआत टैग में शामिल है, इसलिए img टैग को एक समाप्त टैग की आवश्यकता नहीं है।

जब आप HTML के साथ एक वेब पेज लिखते हैं, तो आप सीधे उस दस्तावेज़ में जावास्क्रिप्ट कोड शामिल कर सकते हैं, या आप जावास्क्रिप्ट कोड फ़ाइल (जो अंत में समाप्त हो सकती है) को संदर्भित कर सकते हैं। HTML दस्तावेज़ से .js)। किसी भी तरह से, आपके दर्शक का वेब ब्राउज़र जावास्क्रिप्ट कोड डाउनलोड करेगा और उपयोगकर्ता द्वारा एक्सेस करने पर इसे चलाएगा

वेब पेज जिसमें जावासक्रिप्ट है। क्लाइंट–साइड जावासक्रिप्ट आपके उपयोगकर्ताओं के वेब ब्राउज़र के अंदर चलता है।

CSS3

कैस्केडिंग स्टाइल शीट्स (सीएसएस) वह भाषा है जिसका उपयोग वेब पृष्ठों में प्रारूपण और विभिन्न लेआउट जोड़ने के लिए किया जाता है। शब्द शैली, जब सीएसएस में उपयोग किया जाता है, तो उपयोगकर्ता को HTML दस्तावेज़ कैसे प्रस्तुत किया जाता है, सहित कई पहलुओं को संदर्भित करता है

✓Typefaces (or font faces)

✓ Type size

✓ Colors

✓ Arrangement of elements in the browser window

✓ Sizes of elements

✓ Borders

✓ Backgrounds

✓ Creation of rounded corners on element borders

जावासक्रिप्ट की तरह, सीएसएस को या तो सीधे एक HTML दस्तावेज़ में रखा जा सकता है, या इसे HTML दस्तावेज़ से जोड़ा जा सकता है। एक बार डाउनलोड होने के बाद, यह तुरंत अपनी बात करता है और आपके विनिर्देशों के अनुसार दस्तावेज़ को प्रारूपित करता है।

CSS में शैली पत्रक (**Style sheets**) CSS नियमों से बने होते हैं, जिनमें ऐसे गुण और मूल्य होते हैं जिन्हें किसी element या elements के समूह पर लागू किया जाना चाहिए। यहां सीएसएस नियम का एक उदाहरण दिया गया है:

```
p{font-size: 14px; font-color: black; font-family: Arial,
sans-serif}
```

यह नियम, बाएं से दाएं पढ़ना, निर्दिष्ट करता है कि सभी पी एलिमेंट्स

(जो HTML में पैराग्राफ को इंगित करते हैं) को टेक्स्ट में प्रदर्शित किया जाना चाहिए: 14px बड़ा, काले, और एरियल फ़ॉन्ट का उपयोग करना। यदि एरियल उपयोगकर्ता के कंप्यूटर पर उपलब्ध नहीं है, तो इसे कुछ सैंस सेरिफ़ टाइपफेस में प्रदर्शित किया जाना चाहिए।

CSS नियम का वह हिस्सा जो curly brackets के बाहर है, चयनकर्ता कहलाता है। यह उन elements का चयन करता है जो curly ब्रैकेट के भीतर के गुणों पर लागू होते हैं।

इस पुस्तक के दौरान, आपको पता चलता है कि HTML और CSS के साथ जावासक्रिप्ट का उपयोग कैसे किया जाता है। हम आपको HTML और CSS कैसे काम करते हैं, यह दिखाने के लिए यहाँ पर्याप्त जानकारी प्रदान करते हैं। यदि आपको अधिक जानने की आवश्यकता है, तो आप उनके बारे में कुछ उत्कृष्ट पुस्तकें पा सकते हैं।

JavaScript is powerful!

वेब ब्राउजर में चलने वाले जावासक्रिप्ट धीमी गति से चलते थे, और प्रोग्रामर के बीच में जावासक्रिप्ट को जल्दी खराब प्रतिष्ठा मिली। आज, जावासक्रिप्ट कोड Compiled कोड के रूप में तेजी से 80 प्रतिशत चलाता है। और, यह हर समय तेज होता रहता है। इसका मतलब यह है कि आज का जावासक्रिप्ट कुछ साल पहले के जावासक्रिप्ट से कहीं अधिक शक्तिशाली है। और, यह उस जावासक्रिप्ट से कई गुना अधिक शक्तिशाली है जिसे पहली बार 1995 पेश किया गया था।

JavaScript is in demand

जावासक्रिप्ट न केवल सबसे व्यापक रूप से ज्ञात प्रोग्रामिंग भाषा है, यह नौकरी के बाजार में सबसे अधिक मांग वाले कौशल में से एक है। यह अनुमान लगाया गया है कि जावासक्रिप्ट प्रोग्रामर के लिए नौकरी बाजार में 2010 और 2020 के बीच 22 प्रतिशत की वृद्धि होगी। रोमांचक चीजें जावासक्रिप्ट के साथ हो रही हैं, और इसे सीखने के लिए पहले से बेहतर

समय कभी नहीं रहा है।

CHAPTER 2

WRITING YOUR FIRST JAVASCRIPT PROGRAM

सरल जावासक्रिप्ट प्रोग्रामिंग समझना मुश्किल नहीं है। इस अध्याय में, आप जावासक्रिप्ट लिखने के लिए अपना कंप्यूटर सेट करने की प्रक्रिया से गुजरते हैं। आप अपना पहला जावासक्रिप्ट प्रोग्राम भी लिखते हैं और प्रोग्रामर के रूप में अपने भविष्य में जावासक्रिप्ट के साथ जो कुछ भी करते हैं, उसके पीछे के मूल सिंटैक्स को जान सकते हैं।

Setting Up Your Development Environment

अपने पहले जावासक्रिप्ट प्रोग्राम को लिखने के लिए शुरुआत से पहले अपने सभी उपकरणों को स्थापित करना महत्वपूर्ण है। हम आपके पसंदीदा जावासक्रिप्ट विकास टूल को डाउनलोड करने और स्थापित करने की प्रक्रिया से चलते हैं, जो संयोगवश, हम इस पुस्तक में उपयोग करते हैं। यदि आपके पास ऐसे ही उपकरण हैं जिन्हें आप पसंद करते हैं, तो कृपया उन का उपयोग करने के लिए स्वतंत्र महसूस करें। हालाँकि, हम अनुशंसा करते हैं कि आप अभी भी पुस्तक के इस भाग को पढ़ने के लिए सीखें कि हमने इन उपकरणों को क्यों चुना है और इनका उपयोग करने के बारे में अपने निर्णय लेने के लिए।

आप प्रत्येक उपकरण को स्थापित करने के बाद, हम उनमें से प्रत्येक से सबसे अधिक लाभ उठाने के लिए आपके साथ कुछ युक्तियां और युक्तियां साझा करते हैं।

Downloading and installing Chrome

जावासक्रिप्ट के साथ काम करते समय हम जिस वेब ब्राउज़र का उपयोग करना पसंद करते हैं, वह Google Chrome है। यदि आप दिन–प्रतिदिन एक अलग वेब ब्राउज़र का उपयोग करना पसंद करते हैं, तो यह ठीक है।

सभी ब्राउज़र जावासक्रिप्ट को बहुत तेज़ी से और सही तरीके से चलाएंगे। हालाँकि, इस पुस्तक के कुछ निर्देश Google Chrome के लिए विशिष्ट होंगे, इसलिए हम अनुशंसा करते हैं कि आप कम से कम इस अध्याय

में अपने कंप्यूटर पर इसे स्थापित करने की प्रक्रिया से गुजरें। हमने इस पुस्तक में Google Chrome का उपयोग करना चुना क्योंकि यह जावास्क्रिप्ट प्रोग्रामर के काम को आसान बनाने के लिए उत्कृष्ट उपकरण प्रदान करता है और क्योंकि यह वर्तमान में इंटरनेट पर सबसे अधिक उपयोग किया जाने वाला वेब ब्राउज़र है। (हां, यह इंटरनेट एक्सप्लोरर से भी अधिक लोकप्रिय है।)

यदि आपके पास Chrome इंस्टॉल नहीं है, तो इसे स्थापित करने के लिए इन चरणों का पालन करें:

1. www.google.com / chrome पर जाएं।

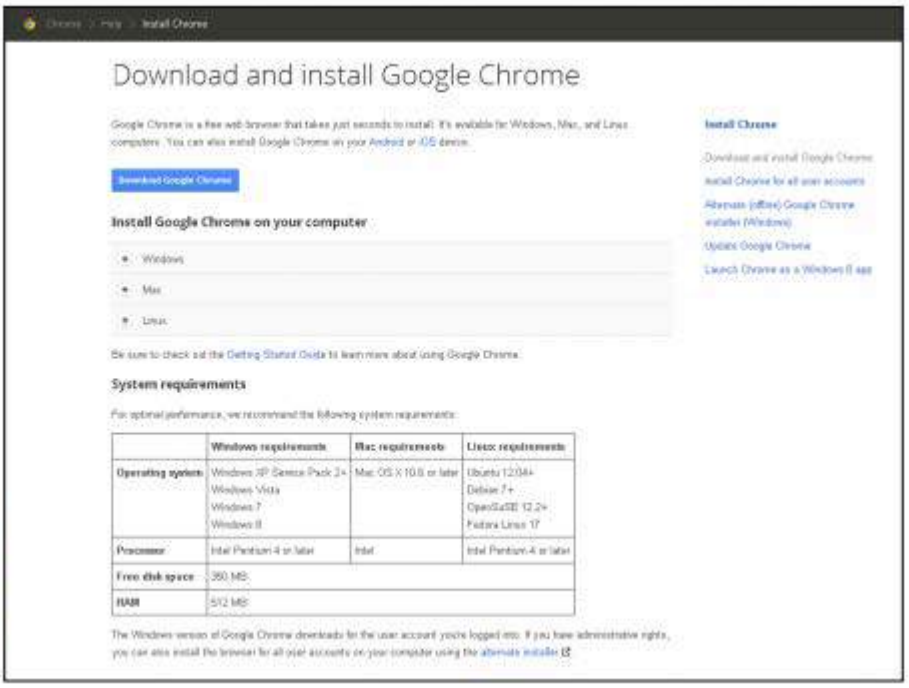


Figure 2-1: Installing Chrome is easy on either Mac or Windows.

2. डाउनलोड टैब पर होवर करें और अपने कंप्यूटर के लिए उपयुक्त संस्करण (version) चुनें।
3. डाउनलोड की गई फ़ाइल खोलें और क्रोम इंस्टॉल करने के लिए निर्देशों का पालन करें।

Now you have a supercharged JavaScript engine!

Google Chrome, जावासक्रिप्ट कोड को पार्स करने, Compiled करने और चलाने के लिए Google V8 के जावासक्रिप्ट इंजन का उपयोग करता है। आप किसके बेंचमार्किंग टेस्ट के आधार पर मानते हैं कि क्रोम ब्राउज़र में जावासक्रिप्ट चलाने का सबसे तेज़ तरीका है, या यह सबसे तेज़ में से एक है। प्रमुख ब्राउज़र निर्माता एक दूसरे से आगे निकलने के लिए लगातार प्रतिस्पर्धा कर रहे हैं।

यह बहुत ज्यादा मायने नहीं रखता है कि वास्तव में किसी एक समय में सबसे तेज कौन है; प्रतियोगिता ने हाल के वर्षों में लीप्स और सीमा द्वारा हर ब्राउज़र के जावासक्रिप्ट इंजन की गति बढ़ा दी है। यदि आप जावासक्रिप्ट प्रदर्शन परीक्षणों में विभिन्न ब्राउज़रों की वास्तविक तुलना देखना चाहते हैं, तो आप http://arewefastyet.com पर ऐसा कर सकते हैं। यह साइट, जिसे फ़ायरफ़ॉक्स ब्राउज़र के निर्माता मोज़िला ने बनाए रखा है, स्वचालित रूप से सबसे लोकप्रिय ब्राउज़रों के जावासक्रिप्ट प्रदर्शन की जाँच और रेखांकन करता है और हर दिन कई बार अपडेट किया जाता है।

Downloading and installing a code editor

एक स्रोत कोड संपादक, जिसे आमतौर पर कोड संपादक के रूप में जाना जाता है, एक पाठ संपादक है जो अतिरिक्त कार्यक्षमता के साथ आपको प्रोग्रामिंग कोड लिखने और संपादित करने में मदद करता है। हम जो उपयोग करते हैं वह **Sublime Tex** है। चुनने के लिए कई कोड संपादक हैं, इसलिए यदि आपके पास पहले से कोई पसंदीदा है **(Notepad, Notepad++, etc)** जिसे आप उपयोग करना पसंद करते हैं और आप इसके साथ सहज हैं, तो कृपया इसका उपयोग करें! ए प्रोग्रामर का कोड संपादक एक बहुत ही व्यक्तिगत पसंद है, और कई लोग पाएंगे कि वे केवल एक विशिष्ट के साथ अधिक सहज महसूस करते हैं। यदि आप पाते हैं कि Sublime Text सिर्फ आपकी शैली के अनुकूल नहीं है, तो

तालिका 2–1 कुछ अन्य विकल्पों को सूचीबद्ध करती है।

Table 2-1 Examples of Other Code Editors

Name	*Location*	*Compatible with . . .*
Coda	`http://panic.com/coda`	Mac only
Aptana	`www.aptana.com`	Mac or Windows
Komodo Edit	`www.activestate.com/komodo-edit/downloads`	Mac or Windows

(continued)

Table 2-1 *(continued)*

Name	*Location*	*Compatible with . . .*
Dreamweaver	`http://adobe.com/products/dreamweaver.html`	Mac or Windows
Eclipse	`www.eclipse.org`	Mac or Windows
Notepad++	`http://notepad-plus-plus.org`	Windows only
TextMate	`http://macromates.com`	Mac only
BBEdit	`www.barebones.com/products/bbedit`	Mac only
EMacs	`www.gnu.org/software/emacs`	Mac or Windows
TextPad	`www.textpad.com`	Windows only
vim	`www.vim.org`	Mac or Windows
Netbeans	`https://netbeans.org`	Mac or Windows

हम Sublime Text का उपयोग करते हैं (चित्र 2–2 देखें इस पुस्तक के लिए क्योंकि यह जावास्क्रिप्ट प्रोग्रामर्स के बीच लोकप्रिय है, और यह अधिक उन्नत प्रोग्रामिंग कार्यों को संभालने के लिए बड़ी संख्या में प्लगइन्स के साथ एक सरल उपयोगकर्ता इंटरफ़ेस प्रदान करता है क्योंकि आप अधिक प्रोग्रामिंग अनुभव प्राप्त करते हैं।

Sublime Text स्थापित करने के लिए, इन चरणों का पालन करें:

1- http://sublimetext.com पर जाएं और अपने ऑपरेटिंग सिस्टम के लिए उपयुक्त संस्करण चुनें।

Figure 2-2: Sublime Text is a seemingly simple-looking text editor with a lot of powerful features.

2- डाउनलोड की गई फ़ाइल को खोलें और Sublime पाठ को स्थापित करने के निर्देशों का पालन करें।

Getting started with Sublime Text

जब आप पहली बार Sublime Text खोलते हैं, तो आप उस पर एक कर्सर के साथ एक साधारण रिक्त पृष्ठ देखते हैं (चित्र 2-3 देखें)।

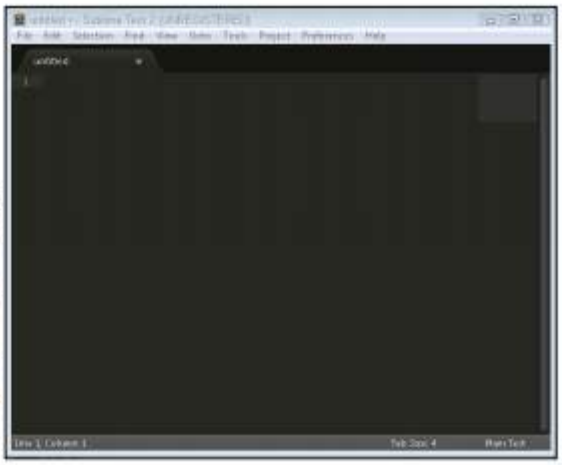


Figure 2-3: The initial Sublime Text user interface. How's that for simplicity?

यदि आपने Sublime Text का उपयोग किया है, तो आप बाईं ओर एक साइडबार देख सकते हैं, जैसा कि दिखाया गया है चित्र 2–4. यदि आपने एक बनाया है, तो यह साइडबार आपकी खुली फाइलों और आपके प्रोजेक्ट की फाइलों को दिखाता है। साइडबार उपयोगी है, और हम अनुशंसा करते हैं कि आपके पास यह खुला है।

साइडबार खोलने के लिए, View⇨SidebarhowShow साइडबार पर क्लिक करें।

अपनी पहली Sublime Text परियोजना फ़ाइल के साथ आरंभ करने के लिए, इन चरणों का पालन करें:

1- इस रूप में File - Save चुनें।

   सहेजें संवाद बॉक्स दिखाई देता है, और आपका डिफ़ॉल्ट सहेजें स्थान दिखाया गया है। यदि आप इस फ़ोल्डर में अपना कोड स्टोर करने से खुश हैं (सबसे अधिक expressions है कि दस्तावेज़ फ़ोल्डर [OSX पर] या मेरा दस्तावेज़ फ़ोल्डर [Windows]), तो चरण पर जाएं। अन्यथा, अपने स्थान पर किसी अन्य स्थान पर नेविगेट करें कंप्यूटर जहां आप अपनी कोड फ़ाइलों को संग्रहीत करना चाहते हैं।

2- एक नया फ़ोल्डर बनाएं और फ़ोल्डर का नाम दें।

3- इस रूप में सहेजें क्षेत्र में, इस पहली फ़ाइल को एक नाम दें और फिर सहेजें पर क्लिक करें। नया फ़ाइल नाम साइडबार में दिखाई देता है और खुले टैब पर नाम आपके चयनित नाम में बदल जाता है।

4- Project - Save प्रोजेक्ट चुनें और आपके द्वारा बनाए गए फ़ोल्डर के अंदर Sublime Text प्रोजेक्ट फ़ाइल को सहेजें। Sublime Text प्रोजेक्ट फाइलें वह होती हैं, जहां Sublime Text की जानकारी संग्रहीत होती है फ़ाइलों और फ़ोल्डरों के बारे में एक परियोजना के साथ जुड़े रहे हैं। प्रोजेक्ट फ़ोल्डर बनाने से आप अपने प्रोग्राम

के सभी विभिन्न प्रकारों को बेहतर तरीके से व्यवस्थित रख सकते हैं।

5- Project - theAdd Folder to Project चुनें, चरण 1 में आपके द्वारा बनाए गए फ़ोल्डर का चयन करें और फिर खोलें क्लिक करें।

साइडबार में एक नई समाप्त सूची दिखाई देती है, जिसे फोल्डर्स कहा जाता है, और आपका फ़ोल्डर, इसकी सामग्री के साथ (परियोजना फ़ाइल और MyFirstProgram.html सहित), इसे नीचे सूचीबद्ध किया जाएगा, जैसा कि चित्र 2–5 में दिखाया गया है।

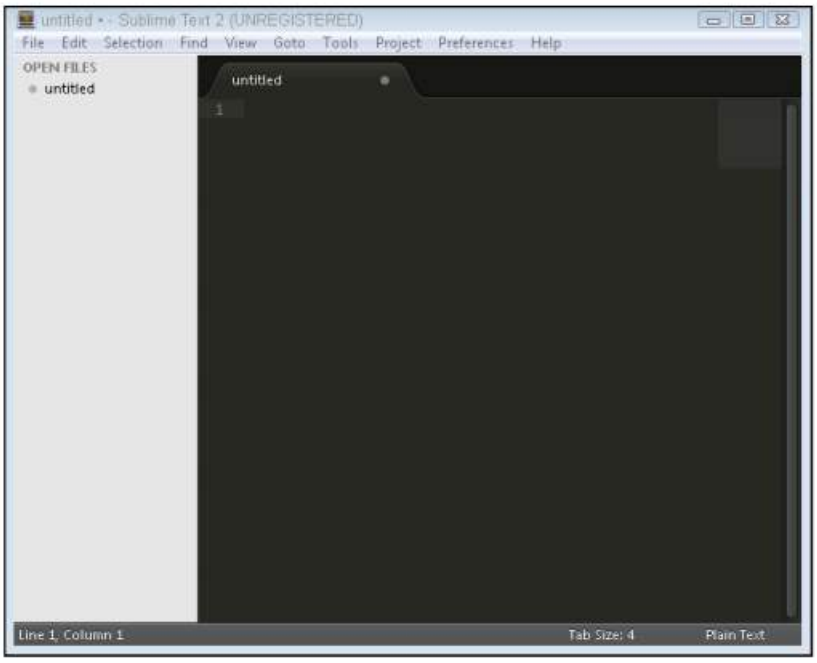


Figure 2-4: Sublime Text with the sidebar open.

आपको अपनी सभी फ़ाइलों और फ़ोल्डरों को व्यवस्थित रखने के लिए, हम आपको अपनी फ़ाइलों और फ़ोल्डरों के नाम के अनुसार कुछ सिफारिशें प्रदान करते हैं। उदाहरण के लिए, आप अपने नए फ़ोल्डर को चरण 2 MyFirstJavaScriptProject से नाम दे सकते हैं। चरण 3 MyFirstProgram में फ़ाइल, और चरण 4 myFirstProject से प्रोजेक्ट।

Choosing a syntax color scheme

Sublime Text syntax रंग उस कोड के प्रकार पर आधारित होते हैं जो आप लिख रहे हैं और फ़ाइल एक्सटेंशन। निम्न HTML और जावास्क्रिप्ट कोड इनपुट 2–1 में दिखाए गए फ़ाइल में दिखाया गया है जिसे आपने डिफ़ॉल्ट रंग योजना देखने के लिए बनाया है।

जैसा कि आप यह पता लगाने वाले हैं कि जावास्क्रिप्ट फ़िंक है। सुनिश्चित करें कि आप सब कुछ ठीक उसी तरह कैपिटल और स्पेल करते हैं जैसा वह लिस्टिंग में है, या आपकी सक्रिप्ट सही तरीके से या बिल्कुल भी काम नहीं कर सकती है। लिस्टिंग 2–1: एक नमूना HTML फ़ाइल जिसमें जावास्क्रिप्ट शामिल है।

Listing 2-1: A Sample HTML File Containing JavaScript

```
<!DOCTYPE html>
<html>
<head>
  <title>Hello, HTML!</title>
  <script>
    function countToTen(){
      var count = 0;
      while (count < 10) {
        count++;
        document.getElementById("theCount").innerHTML +=
           count + "<br>";
      }
    }
```

(continued)

Listing 2-1 *(continued)*

```
Code-Line Before Listing Code  </script>
</head>
<body onload="countToTen();">
  <h1>Let's Count to 10 with JavaScript!</h1>
  <p id="theCount"></p>
</body>
</html>
```

चित्र 2–6 दिखाता है कि फ़ाइल हमारे लिए Sublime Text में कैसी दिखती है।

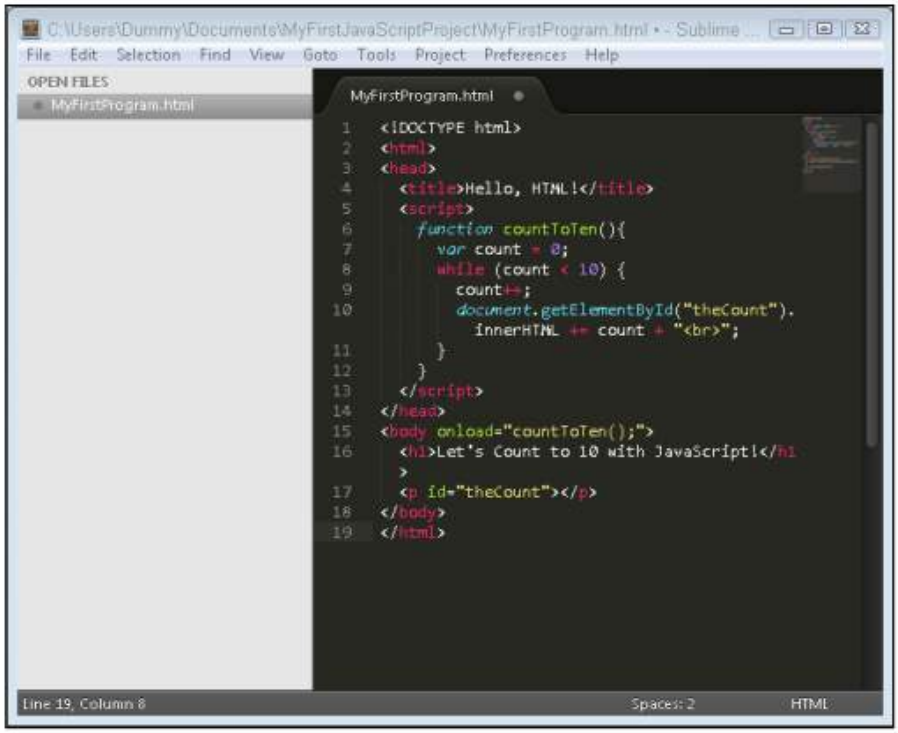


Figure 2-6: Sublime Text applies colors to all of the different parts of your code.

यदि आपको वर्तमान में चुनी गई रंग योजना पसंद नहीं है, तो आप PreferencesolorColor योजना चुनकर और फिर अन्य रंग योजना चुनकर इसे बदल सकते हैं।

अन्य रंग योजनाओं में से कुछ को आज़माएँ और एक को आप पसंद करें। इस पुस्तक के लिए हम जो उपयोग करते हैं उसे मोनोकई ब्राइट कहते हैं।

यदि आप अपने द्वारा लिखे गए प्रोग्राम को आज़माना चाहते हैं, तो इन चरणों का पालन करें:

1- File⇨Save चुनकर फ़ाइल को सहेजें।

2- अपना क्रोम ब्राउज़र खोलें और Ctrl O दबाएं। एक ओपन फ़ाइल विंडो दिखाई देती है।

3- अपने कंप्यूटर पर फ़ाइल पर नेविगेट करें और इसे चुनें।

4- ओपन बटन पर क्लिक करें। आपके ब्राउज़र में फाइल खुल जाएगी।

आपके ब्राउज़र को चित्र 2–7 जैसा दिखना चाहिए यदि यह नहीं है, तो बहुत सावधानी से अपने कोड की जांच करें – आपके पास शायद एक छोटा टाइपो है। कोई भी बदलाव करने के बाद अपनी फ़ाइल को सहेजना न भूलें!

Figure 2-7: Running a simple counting program in Chrome.

आप कमांड **command + S (on the Mac)** को दबाकर या **Control + S (On Windows)** अपनी फ़ाइल को भी सहेज सकते हैं । एक बार जब आप उनके साथ कुशल हो जाते हैं, तो कीबोर्ड शॉर्टकट आपको बहुत समय बचाएंगे।

Some helpful Sublime Text shortcuts

Sublime Text एक साधारण पाठ संपादक (**text editor**) की तरह दिखता है, लेकिन मूर्ख मत बनो! एक मास्टर प्रोग्रामर का एक सही निशान उसकी या उसके कीबोर्ड शॉर्टकट्स का उपयोग करके कोड को क्रैंक करने और जल्द से जल्द संपादन करने की क्षमता है। तालिका 2–2 कुछ सूचीबद्ध करती है।

Sublime Text प्रदान करता है कि कई कीबोर्ड शॉर्टकट। इनका अभ्यास करें, और आप अपने दोस्तों और सहकर्मियों को अपने सुपर–कुलीन

कौशल के साथ जल्दी से प्रभावित कर पाएंगे।

Table 2-2 Commonly Used Sublime Text Editing Keyboard Shortcodes

Mac	*Windows*	*Description*
Command+X	Ctrl+X	Delete line
Command+Return	Ctrl+Enter	Insert line after
Command+Shift+Return	Ctrl+Shift+Enter	Insert line before
Command+Control+Up Arrow	Ctrl+Shift+Up Arrow	Move line/Selection Up
Command+Control+Down Arrow	Ctrl+Shift+Down Arrow	Move line/Selection down

(continued)

Table 2-2 *(continued)*

Mac	*Windows*	*Description*
Command+L	Ctrl+L	Select line; repeat to select next lines
Command+D	Ctrl+D	Select word; repeat to select other occurrences
Control+M	Ctrl+M	Jump to closing parentheses; repeat to jump to opening parentheses
Control+Shift+M	Ctrl+Shift+M	Select all contents of current parentheses
Command+K+Command+K	Ctrl+k+k	Delete from cursor to end of line
Command+K+Delete	Ctrl+K+Delete	Delete from cursor to beginning of line
Command+]	Ctrl+]	Indent current line(s)
Command+[	Ctrl+[	Un-indent current line(s)
Command+Shift+D	Ctrl+Shift+D	Duplicate line(s)
Command+J	Ctrl+J	Join line below to the end of the current line
Command+/	Ctrl+/	Comment/un-comment current line
Command+Option+/	Ctrl+Shift+/	Block comment current selection

Command+/	Ctrl+/	Comment/un-comment current line
Command+Option+/	Ctrl+Shift+/	Block comment current selection
Command+Y	Ctrl+Y	Redo or repeat last key-board shortcut command
Command+Shift+V	Ctrl+Shift+V	Paste and indent correctly
Control+Space	Ctrl+Space	Select next auto-complete selection
Control+U	Ctrl+U	Soft Undo; jumps to your last change before undoing change when repeated
Control+Shift+Up	Ctrl+Alt+Up	Column selection up
Control+Shift+Down	Ctrl+Alt+Down	Column selection down
Control+Shift+W	Alt+Shift+W	Wrap selection in html tag

Reading JavaScript Code

इससे पहले कि आप जावास्क्रिप्ट प्रोग्राम लिखना शुरू करें, आपको जावास्क्रिप्ट के कुछ नियमों के बारे में पता होना चाहिए:

- जावास्क्रिप्ट case-sensitive है। हम इसे पूरी पुस्तक में कई बार दोहराते हैं, क्योंकि यह एक त्रुटि है कि जो लोग जावास्क्रिप्ट में नए हैं, वे अक्सर बनाते हैं। जावास्क्रिप्ट के लिए, पैंट और पैंट शब्द पूरी तरह से अलग हैं।

- जावास्क्रिप्ट White Space के बारे में ज्यादा परवाह नहीं करता है। सफेद स्थान में रिक्त स्थान, टैब और लाइन ब्रेक शामिल हैं – कोई भी वर्ण जिसमें दृश्य प्रतिनिधित्व नहीं है। जब आप जावास्क्रिप्ट कोड लिख रहे होते हैं, तो इससे कोई फर्क नहीं पड़ता कि आप कोड के भीतर एक स्थान, दो स्थान, एक टैब, या यहां तक कि एक लाइन ब्रेक (ज्यादातर मामलों में) का उपयोग करते हैं। जावास्क्रिप्ट White Space की उपेक्षा करेगा। एक अपवाद तब होता है जब आप पाठ को लिख रहे होते हैं, जिसे आप जावास्क्रिप्ट से स्क्रीन पर प्रिंट करना चाहते हैं। इस स्थिति में, आपके द्वारा उपयोग किया जाने वाला सफेद स्थान अंतिम परिणाम में दिखाई

देगा। आपके कोड में सफेद स्थान के संबंध में सबसे अच्छा अभ्यास, पर्याप्त स्थान का उपयोग करना है जो आपके कोड को पढ़ने में आसान है और यह भी संगत है कि आप इस स्थान का उपयोग कैसे करते हैं।

- आरक्षित शब्दों के लिए बाहर देखो। जावास्क्रिप्ट में उन शब्दों की एक सूची होती है जिनका भाषा के लिए विशेष अर्थ होता है। हम इन शब्दों को अध्याय 3 में सूचीबद्ध करते हैं। अभी के लिए, बस इस बात का ध्यान रखें कि कुछ शब्द, जैसे कि फ़ंक्शन, जबकि, ब्रेक और विशेष अर्थ हैं।

- जावास्क्रिप्ट को अर्धविराम (**semicolons**) पसंद हैं: जावास्क्रिप्ट कोड बयानों से बना है। आप वाक्यों के समान बयानों के बारे में सोच सकते हैं। वे उसी तरह जावास्क्रिप्ट प्रोग्राम के लिए मौलिक बिल्डिंग ब्लॉक हैं जैसे वाक्य पैराग्राफ के बिल्डिंग ब्लॉक हैं। जावास्क्रिप्ट में, बयान एक अर्धविराम के साथ समाप्त होते हैं।

  यदि आप एक बयान के अंत में एक अर्धविराम का उपयोग नहीं करते हैं, तो जावास्क्रिप्ट इसे आपके लिए रखेगा। हालांकि, अप्रत्याशित परिणाम हो सकते हैं, इसलिए इसे अर्धविराम के साथ हमेशा बयानों को समाप्त करने का सबसे अच्छा अभ्यास माना जाता है।

Running JavaScript in the Browser Window

हालांकि यह कई अलग–अलग वातावरणों में देखा जाता है, लेकिन जंगली ब्राउज़रों में जावास्क्रिप्ट देखने के लिए सबसे आम जगह वेब ब्राउज़र में चल रही है। इनपुट और आउटपुट को नियंत्रित करना, वेब पेजों में हेरफेर करना, क्लिक और स्क्रॉल जैसे सामान्य ब्राउज़र इवेंट्स को हैंडल करना और वेब ब्राउज़र की विभिन्न विशेषताओं को नियंत्रित करना वही है जो जावास्क्रिप्ट करने के लिए पैदा हुआ था!

एक वेब ब्राउज़र में जावास्क्रिप्ट चलाने के लिए, आपके पास तीन विकल्प होंगे, जिनमें से सभी होगा निम्नलिखित पृष्ठों में दिखाया गया है:

- Attribute इसे सीधे HTML ईवेंट विशेषता में रखें
- इसे सक्रिप्ट खोलने और बंद करने वाले टैग के बीच रखें
- इसे एक अलग दस्तावेज़ में रखें और इसे अपने HTML दस्तावेज़ में शामिल करें।

कई बार, आप किसी एक वेब पेज के भीतर सभी तीन तकनीकों के संयोजन का उपयोग करेंगे। हालांकि, यह जानना कि प्रत्येक का उपयोग कब करना महत्वपूर्ण है और एक कौशल है जिसे आप अधिक अभ्यास के साथ सीखेंगे।

Using JavaScript in an HTML event attribute

HTML में कई विशेष विशेषताएं हैं जो वेब ब्राउज़र में कुछ होने पर या उपयोगकर्ता द्वारा कुछ करने पर जावास्क्रिप्ट को ट्रिगर करने के लिए डिज़ाइन की जाती हैं। यहां एक HTML विशेषता का एक उदाहरण दिया गया है जिसमें एक घटना विशेषता है जो माउस क्लिक घटनाओं के प्रति प्रतिक्रिया करता है:

```
<button id="bigButton" onclick="alert('Hello
          World!');">Click Here</button>
```

इस मामले में, जब कोई उपयोगकर्ता इस HTML element द्वारा बनाए गए बटन पर क्लिक करता है, तो हैलो व र्ल्ड शब्द के साथ एक पॉपअप दिखाई देगा!.

HTML में 70 से अधिक विभिन्न ईवेंट विशेषताएँ हैं। तालिका 2–3 सबसे अधिक इस्तेमाल होने वाले लोगों को दिखाती है।

Table 2-3	Commonly Used HTML Event Attributes
Attribute	***Description***
`onload`	Runs the script after the pages finishes loading
`onfocus`	Runs the script when the element gets focus (such as when a text box is active)
`onblur`	Runs the script when the element loses focus (such as when the user clicks a new text box in a form)
`onchange`	Runs the script when the value of an element is changed

Attribute	***Description***
`onselect`	Runs the script when text has been submitted
`onsubmit`	Runs the script when a form has been submitted
`onkeydown`	Runs the script when a user is pressing a key
`onkeypress`	Runs the script when a user presses a key
`onkeyup`	Runs the script when a user releases a key
`onclick`	Runs the script when a user mouse clicks an element
`ondrag`	Runs the script when an element is dragged
`ondrop`	Runs the script when a dragged element is being dropped
`onmouseover`	Runs the script when a user moves a mouse pointer over an element

यद्यपि वे उपयोग करने में आसान हैं, लेकिन ईवेंट विशेषताओं का उपयोग करना वास्तव में कई जावास्क्रिप्ट प्रोग्रामर द्वारा कम–से–आदर्श अभ्यास माना जाता है। हम उन्हें इस पुस्तक में प्रदर्शित करते हैं क्योंकि वे बहुत व्यापक रूप से उपयोग किए जाते हैं और सीखना आसान है। हालांकि, अभी के लिए, बस इस बात से अवगत रहें कि जावास्क्रिप्ट कोड लिखने का एक बेहतर तरीका है जो इवेंट विशेषताओं का उपयोग करने की तुलना में घटनाओं पर प्रतिक्रिया करता है। हम इस बेहतर तरीके को अध्याय 11 में शामिल करते हैं।

Using JavaScript in a script element

HTML सक्रिप्ट element आपको जावास्क्रिप्ट को एक HTML दस्तावेज़ में एम्बेड करने की अनुमति देता है। अक्सर सक्रिप्ट elements को मुख्य element के भीतर रखा जाता है, और वास्तव में, इस प्लेसमेंट को अक्सर आवश्यकता के रूप में कहा जाता था। आज, हालांकि, सक्रिप्ट elements

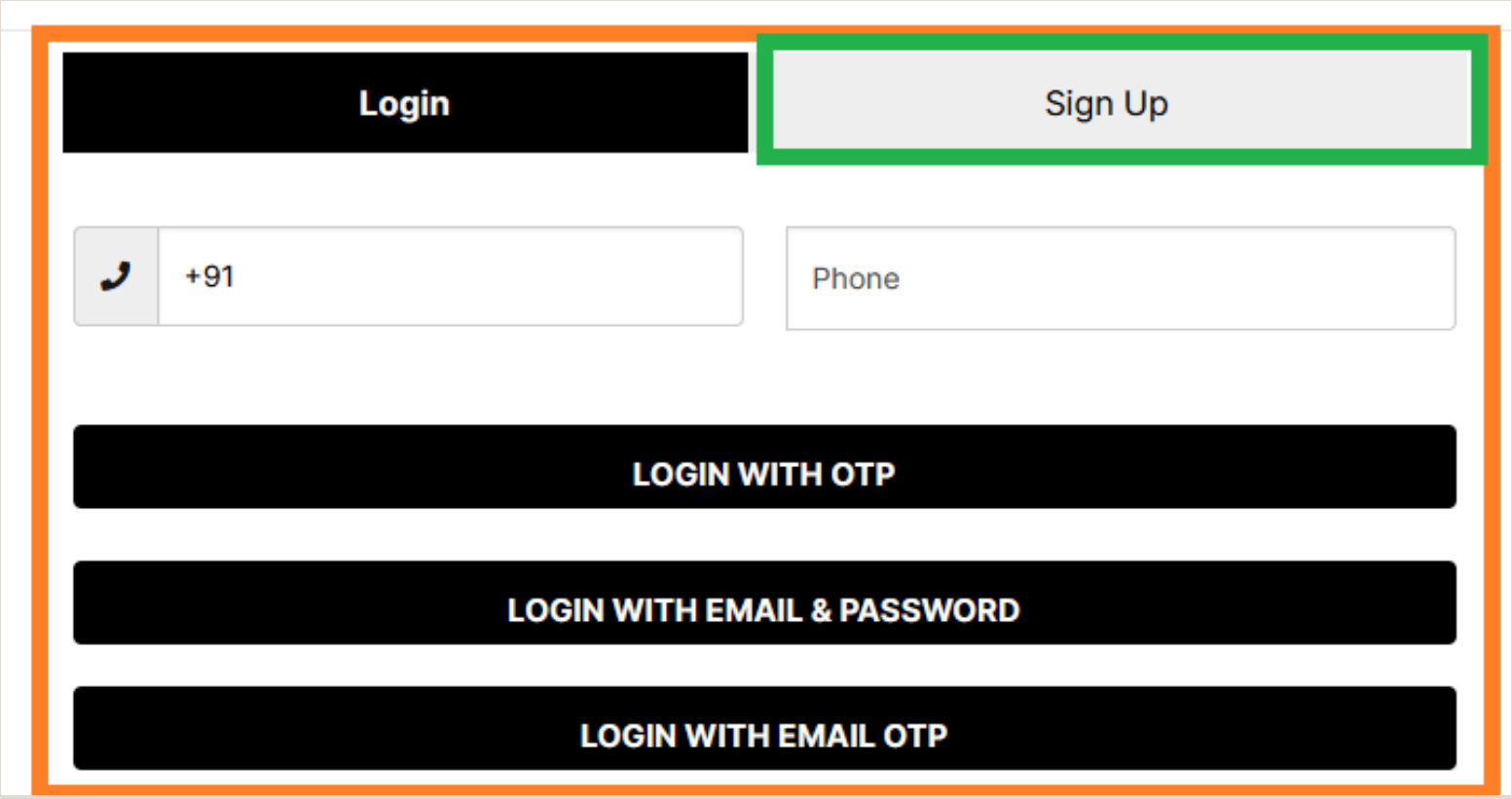
HOW TO GET COMPLETE PDF EBOOK

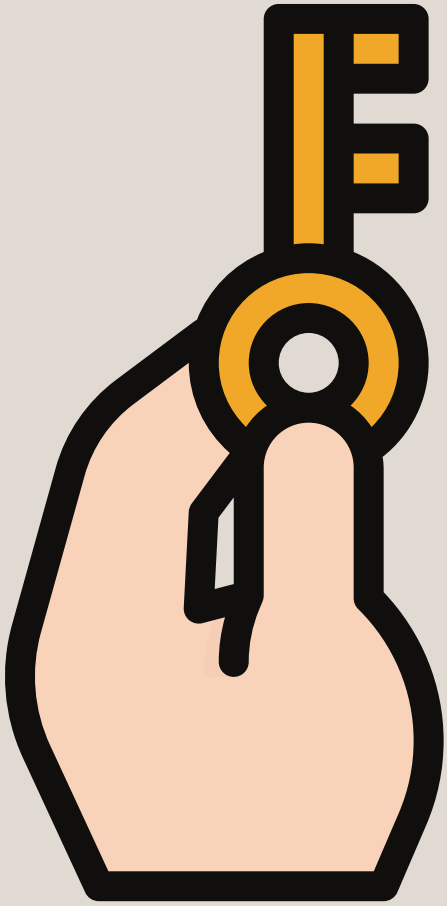
→BUY NOW←

Pay Paytm

कोई भी **2 Ebooks** खरीदने पर **20%** डिस्काउंट- **India20**

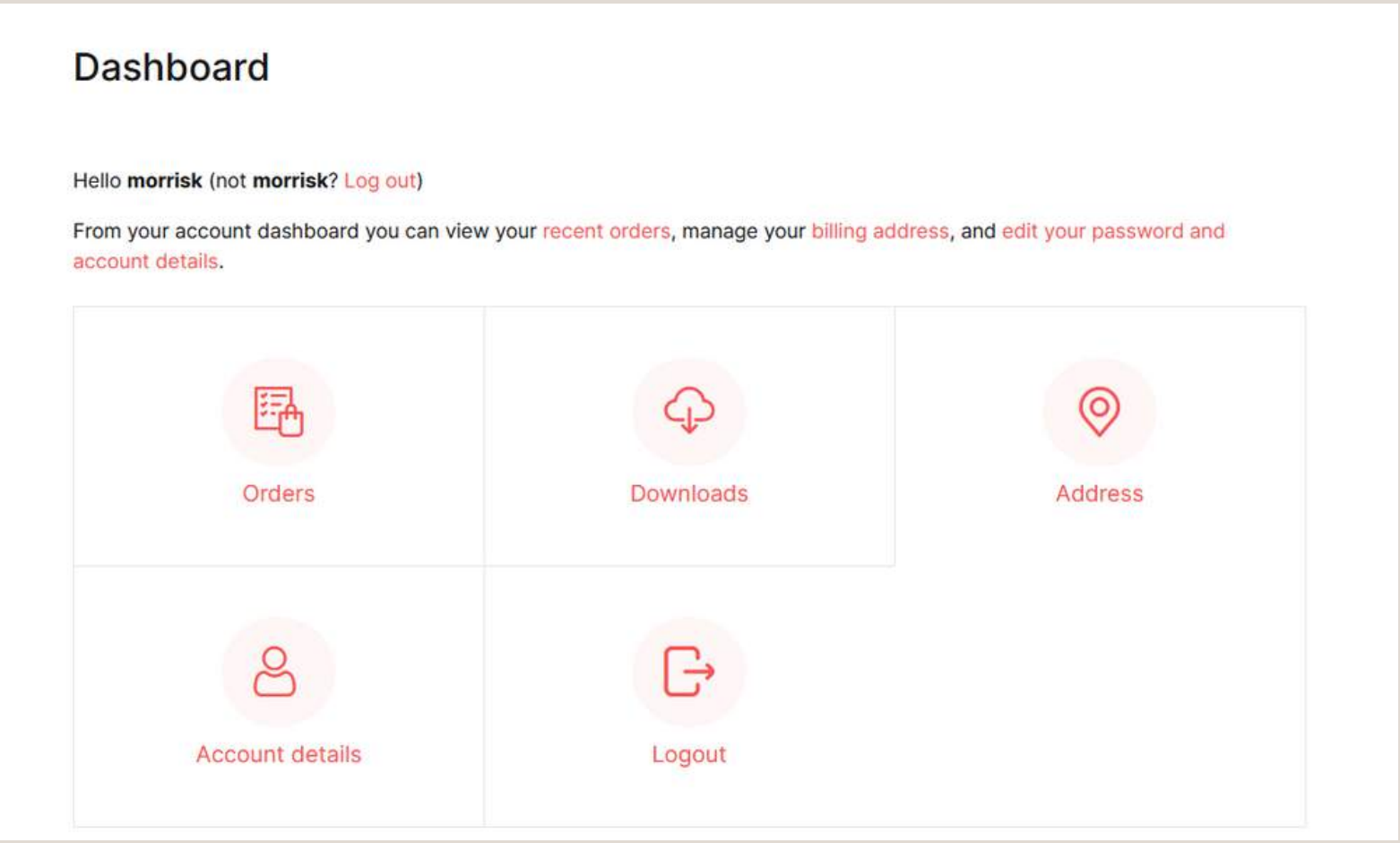
Buy Now पर क्लिक करने पर आपको **चेकआउट पेज** पर लॉगिन फॉर्म पर पहिले ईमेल से लॉगिन करना होगा।





Login Succesfull के बाद आपको पेमेंट करके Ebooks वही से डाउनलोड कर सकते है।

उसके बाद आप **My Account** पेज पर जाकर **Download** में Ebooks डाउनलोड कर सकते है।



Thank you